Non-Linear Discriminant Analysis

Gaston BAUDAT, Fatiha ANOUAR

MEI, Mars Electronics International
1301 Wilson Drive, West Chester, PA 19380, USA

Email: gaston.baudat@eu.effem.com  Phone: + 1 610 430 27 51
Email: fatiha.anouar@effem.com  Phone: + 1 610 430 25 22
Fax: + 1 610 430 27 95
Presentation plan

- Kernel Trick
- Linear Discriminant Analysis (LDA).
- Generalized Discriminant Analysis (GDA).
- Feature Vector Selection (FVS).
- Sparse GDA using the FVS approach.
- Conclusions.
- Some references.
Kernel trick & dot product

Let $\phi(\vec{x}_i)$ be an operator which maps data from an input space $X$ into a feature space $F$:

\[ \phi(\vec{x}_i) \]
Kernel trick & dot product

Ex: Polynomial mapping

- As an example assume the following mapping (2D -> 3D):

  \[
  \phi(\vec{x}) = \begin{pmatrix}
  \varphi_{x,1} \\
  \varphi_{x,2} \\
  \varphi_{x,3}
  \end{pmatrix} = \begin{pmatrix}
  x_1^2 \\
  x_2^2 \\
  \sqrt{2 \cdot x_1 \cdot x_2}
  \end{pmatrix} \quad \vec{x} = \begin{pmatrix}
  x_1 \\
  x_2
  \end{pmatrix}
  \]
Kernel trick & dot product

Ex: Dot product in F

- Explicit dot product in F:

\[ \phi^T (\vec{x}) \cdot \phi(\vec{y}) = \phi_{x,1} \phi_{y,1} + \phi_{x,2} \phi_{y,2} + \phi_{x,3} \phi_{y,3} \]

- Implicit dot product in F:

\[ \Rightarrow x_1^2 y_1^2 + x_2^2 y_2^2 + 2x_1 y_1 x_2 y_2 = (x_1 y_1 + x_2 y_2)^2 \]

\[ \phi^T (\vec{x}) \cdot \phi(\vec{y}) = k(\vec{x}, \vec{y}) = (\vec{x}^T \cdot \vec{y})^2 \]
Kernel trick & dot product

Kernel function

- The implicit form of the dot product in $F$ uses a kernel function $k(\tilde{x}, \tilde{y})$.

- $k(\tilde{x}, \tilde{y})$ does not need the evaluation (or knowledge) of $\phi(\tilde{x})$ nor $\phi(\tilde{y})$.

- Any algorithm using only dot products can be expressed implicitly in $F$. 

Kernel trick & dot product
some classical kernels

- **Gaussian:**
  \[ k(\vec{x}, \vec{y}) = \exp\left(-\frac{||\vec{x} - \vec{y}||^2}{\sigma^2}\right) \]
  \( N_F = \infty \)

- **Sigmoid:**
  \[ k(\vec{x}, \vec{y}) = \tanh(a \, \vec{x}^T \cdot \vec{y} + b) \]
  \( N_F = \infty \)

- **Homogeneous polynomial:**
  \[ k(\vec{x}, \vec{y}) = (\vec{x}^T \cdot \vec{y})^d \quad \forall d \in \{1, 2, 3, \ldots\} \]
  \( N_F = \frac{(d + N_X - 1)!}{d! (N_X - 1)!} \)
Linear Discriminant Analysis

LDA

- LDA versus PCA projection:

  Discriminant axis

  1st PCA axis
  (maximum variance)

  LDA axis

  PCA axis
LDA
Classical criterion

Let's assume N clusters and M samples:
- \( C \) : total covariance matrix
- \( G \) : covariance matrix of the centers
- \( \vec{v}_i \) : \( i^{th} \) discriminant axis \((i=1,\ldots,N-1)\)
- LDA maximizes the variance ratio:

\[
\lambda_i = \frac{\text{Inter-class variance}}{\text{Total variance}} \quad \Rightarrow \quad \lambda_i = \frac{\vec{v}_i^T G \vec{v}_i}{\vec{v}_i^T C \vec{v}_i}
\]
LDA
Resolution

- The solution is based on an eigen system:

\[ \lambda_i \vec{v}_i = C^{-1} G \vec{v}_i \]

- The eigen vectors are linear combinations of the learning samples:

\[ \vec{v}_i = \sum_{j=1}^{M} \alpha_{ij} \vec{x}_j \]
Generalized Discriminant Analysis (GDA)

- LDA in the feature space $F$.
- The eigen vectors are linear combinations of the learning samples:

$$\bar{v}_i = \sum_{j=1}^{M} \alpha_{ij} \phi(\bar{x}_j)$$

- Consequently any projection becomes:

$$\bar{v}_i^T \cdot \bar{z} = \sum_{j=1}^{M} \alpha_{ij} k(\bar{x}_j, \bar{z})$$
We assume the data are centered in $F$.

Total covariance matrix:

$$V = \frac{1}{M} \sum_{j=1}^{M} \phi(\bar{x}_j)\phi^T(\bar{x}_j)$$

Covariance matrix of the $N$ centers:

$$B = \frac{1}{M} \sum_{l=1}^{N} n_l \phi_l \phi_l^T$$

$$\phi_l = \frac{1}{n_l} \sum_{k=1}^{n_l} \phi(\bar{x}_{l,k})$$
GDA Resolution

- Let $K$ be the kernel matrix $(M \times M)$:
  \[ K = (k(\overline{x}_i, \overline{x}_j))_{i=1,\ldots,M}^{j=1,\ldots,M} \]

- The LDA criterion in $F$ becomes:
  \[ \lambda_i = \frac{\overline{\alpha}_i^T KWK \overline{\alpha}_i}{\overline{\alpha}_i^T K^2 \overline{\alpha}_i} \]

  where $W$ is a $(M \times M)$ bloc diagonal matrix of weights $1/n_i$. 
Let's use an eigen decomposition of $K$: 

$$K = U \Gamma U^T$$

Then by substitution:

$$\lambda_i = \frac{\tilde{\beta}_i^T U^T W U \tilde{\beta}_i}{\tilde{\beta}_i^T \tilde{\beta}_i} \quad \text{where} \quad \tilde{\beta}_i = \Gamma U^T \tilde{\alpha}_i$$

Finally it is just a classical eigen system:

$$\lambda_i \tilde{\beta}_i = U^T W U \tilde{\beta}_i$$
GDA
An example

Fisher’s iris data (3 clusters, 4D).

LDA

GDA
Gaussian kernel
\( \sigma = 0.7 \)

+ Iris setosa
x Iris versicolor
° Iris virginica

11/14-15/02 CNAM Paris
Gaston Baudat & Fatiha Anouar MEI©
GDA
An application

- Seed classification (SNES-France)
  3 classes: *Medicago sativa* L., *Melilotus sp* & *Medicago lupulina* L

* Gaussian kernel (σ=0.5)

<table>
<thead>
<tr>
<th>Methods</th>
<th>Learning error</th>
<th>Test error</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>27.2%</td>
<td>32.7%</td>
</tr>
<tr>
<td>GDA*</td>
<td>0%</td>
<td>14.9%</td>
</tr>
<tr>
<td>Probabilistic NN</td>
<td>0%</td>
<td>14.4%</td>
</tr>
</tbody>
</table>
GDA

Some comments

- $k(\vec{x}_i, \vec{x}_j) = \vec{x}_i^T \vec{x}_j$ defines the LDA in $X$.

- The $\alpha_i$ coefficients are not unique. One possible solution is:

$$\tilde{\alpha}_i = U \Gamma^{-1} \tilde{\beta}_i$$

- Without special care this leads to a dense expansion for the discriminant axes. Meaning the all $M$ samples are involved.
Often the data spans a subspace in F with a dimension lower than the size M of the learning data.

**Idea:** Describe this subspace by L Feature Vectors (FV) taken among the samples. They define a basis S in F, with L ≤ M.
The FVS is based on a sequential forward selection maximizing a fitness function defined as follows:

\[
J_S = \frac{1}{M} \sum_{i=1}^{M} \frac{\|\hat{\phi}(\vec{x}_i)\|^2}{\|\phi(\vec{x}_i)\|^2}
\]

\[
J_S = \frac{1}{M} \sum_{i=1}^{M} \hat{K}_{i}^{T} K_{SS}^{-1} \hat{K}_{i} k_{ii} - 1
\]

\[
0 \leq J_S \leq 1
\]
FVS
Empirical kernel map

- After the FVS we can project any sample using the basis $S$. This provides new explicit vectors:

$$\tilde{K}_{Si} = (k(\tilde{x}_{S1}, \tilde{x}_i), ..., k(\tilde{x}_{SL}, \tilde{x}_i))^T$$

- This is known as an empirical kernel map.
FVS
Sparse GDA using FVS

- After the FVS and projection we use the LDA to approximate the GDA.

- Then any projection on a discriminant axis uses an expansion of only $L$ terms:

$$\tilde{\mathbf{V}}_i^T \cdot \tilde{\mathbf{z}} = \sum_{j=1}^{L} \alpha_{ij} k(\tilde{x}_{sj}, \tilde{z})$$
FVS-GDA
An example

- 2 clusters (‘o’ & ‘+’). 100 samples to learn and 100 others for testing.

Gaussian kernel
\( \sigma = 0.5 \)

Decision threshold \( L=40 \) / GDA

Test rate versus \( L \)
Conclusions

- The GDA allows reusing the LDA approach for non-linear cases.

- The projection of samples using a non-linear discriminant scheme provides a convenient way to visualize, analyze, and perform other tasks, such as classification with linear methods.

- Sparse techniques such as FVS overcome the cost of a dense expansion for the discriminant axes.
Some references


